OpenSees Workshop
Online Tutoring Course

S. Mojtaba Hosseini G.,
PhD. Structural and Earthquake Engineering
Educational Background:
2004-2008 BSc, Civil Engineering, KNT University
2008-2010 MSc, Earthquake Engineering, Sharif University of Technology
2010-Present PhD, Structural and Earthquake Eng., Sharif University of Technology

Hadi Kenarangi,
PhD. Structural and Earthquake Engineering
Educational Background:
2003-2007 BSc, Civil Engineering, Tabriz University
2007-2009 MSc, Earthquake Engineering, Sharif University of Technology
2013 PhD, Structural and Earthquake Eng., SUNY at Buffalo
Hadi Kenarangi

Sharif University OpenSees Student Group searches for talented volunteers who can contribute in developing new commands, examples, writing tutoring hand outs, developing SOC (SAP2000 to OpenSees Converter developed by this group).

Volunteers are expected to be experienced in OpenSees, MATLAB and Programming Languages. Knowledge in Microsoft Visual Studio will be an advantage.

Please send your CV to Mr. Seyed Mojtaba Hosseini (hosseinimojtaba@gmail.com) or Hadi Kenarangi (hadi.kenarangi@gmail.com).

Thanks!

OpenSees
Sharif University of Technology
Student Group
SUT OpenSees Group

Some of Past works (Since 2007):

• **OpenSees Book:** coming soon
• **OpenSees Tutoring Courses in:**
  • Sharif University of Technology, Tehran, Iran.
  • Khajeh Nasir University of Technology, Tehran, Iran.
  • Civil House Engineering Institute, Tehran, Iran.
  • Noshirvani University of Technology, Babol, Iran.
  • 9th International Congress on Civil Engineering, 2012 May 8-10, Isfahan University Of Technology, Isfahan, Iran.

• **Development of Related Softwares:**
  • SAP2000 to OpenSees Convertor (SOC2D), A code developed in MatLab which easily converts SAP2000 models into OpenSees.
  • SOC3D

• **Academic and Professional Projects:**
  • Modeling various structural models and simulations such as:
    – Bridges, Steel and RC Buildings, Spatial Structures, Passive and Active Control, Masonry Infill Walls, SMA, Wind Turbine, etc.
What is OpenSees?

- A software framework for simulation applications in earthquake engineering using finite element methods. OpenSees is not a code.

- As open-source software, it has the potential for a community code for earthquake engineering.

- OpenSees has been under development by PEER since before 1997.

- Large group of developers and user.

- NEESgrid and NEESit support integration and extension since 2003.

- Open-source and royalty free license for noncommercial use.
OpenSees Framework

- **ModelBuilder**: Constructs the objects in the model and adds them to the Domain.
- **Domain**: Holds the state of the model in t and (t+dt).
- **Recorder**: Monitors user defined parameters in the model during analysis.
- **Analysis**: Moves the state of the Model from t to t+dt.
OpenSees

- **ModelBuilder Object** is responsible for building the objects in the model and adding them to the domain.

- **Recorder Object** monitors user-specified objects of the model during the analysis.

- **Analysis Object** is responsible for performing the analysis.

- **Domain Object** is responsible for storing the objects created by the ModelBuilder object and for providing the Analysis and Recorder objects access to these objects.
Model-Building Objects

- model Command
- node Command
- mass Command
- Constraints Objects
- uniaxialMaterial Command
- nDMaterial Command
- section Command
- element Command
- block Command
- region Command
- Geometric Transformation Command
- Time Series
- pattern Command
Recorder Objects

• Node Recorder
• EnvelopeNode Recorder
• MaxNodeDisp Recorder
• Drift Recorder
• Element Recorder
• EnvelopeElement Recorder
• Display Recorder
• Plot Recorder
• Playback Command
Analysis Objects

- constraints Command
- numberer Command
- analysis Command
- algorithm Command
- integrator Command
- system Command
- test Command
- analyze Command
- rayleigh Command
- eigen Command
- dataBase Commands
A document providing the syntax and description of OpenSees commands in 3 formats:

- **HTML Manual** – on-line HTML document, residing on OpenSees server. Always going to be the most current.

- **MS Word** – downloadable and printable Word document in PDF format.

- **Offline Windows** – downloadable .chm file. It is similar to the HTML format, but the file resides on your computer.
1. HTML On-line Format
Elastic Beam Column Element

This command is used to construct an elasticBeamColumn element object. The arguments for the construction of an elastic beam-column element depend on the dimension of the problem, \( n_{d, m} \) (page 29):

For a two-dimensional problem:

\[
\text{element elasticBeamColumn $\text{NameTag}$ $\text{SiNode}$ $\text{SjNode}$ $\text{Sf}$ $\text{Sg}$ $\text{Sjy}$ $\text{Siz}$ $\text{StransfTag}$}
\]

For a three-dimensional problem:

\[
\text{element elasticBeamColumn $\text{NameTag}$ $\text{SiNode}$ $\text{SjNode}$ $\text{Sf}$ $\text{Sg}$ $\text{Sjy}$ $\text{Siz}$ $\text{StransfTag}$}
\]

- \( \text{NameTag} \): unique element object tag
- \( \text{SiNode} \): end nodes
- \( \text{SjNode} \): end nodes
- \( \text{Sf} \): cross-sectional area of element
- \( \text{Sg} \): Young's Modulus
- \( \text{Sj} \): shear modulus
- \( \text{Siz} \): torsional moment of inertia of cross section
- \( \text{Siz} \): second moment of area about the local z-axis
- \( \text{Siz} \): second moment of area about the local y-axis
- \( \text{StransfTag} \): identifier for previously-defined coordinate transformation (page 280) (CrdTransf) object

The valid queries to an elastic beam-column element when creating an ElementRecorder (page 307) object are 'stiffness' and 'force.'
Open System for Earthquake Engineering Simulation
User Manual

Software Authors: Frank McKenna, Gregory L. Fenves, et al.

Manual Authors: Silvia Mazzoni, Frank McKenna, Michael H. Scott, Gregory L. Fenves, et al.

Pacific Earthquake Engineering Research Center
University of California, Berkeley

version x.

please send questions and comments about the manual to silvia@peer.berkeley.edu
The OpenSees Quick Reference Guide
Opensees Student Group, May 8, 2012
Seyed Mojtaba Hosseini Gelekolai and Hadi Kenarangi
Sharif University of Technology, Tehran, Iran

1-Modeling Commands

model modelBuilderType <specific model builder args>
model BasicBuilder -ndm ndm? <-ndf ndf?>

node nodeTag? (ndm coordinates?) <-mass (ndf values?)>

mass nodeTag? (ndf values?)

uniaxialMaterial materialType <specific material args>
uniaxialMaterial Elastic matTag? E? <eta?>
uniaxialMaterial ElasticPP matTag? E? ep?
uniaxialMaterial Parallel matTag? tag1? tag2? ... <-min min?> <-max max?>
uniaxialMaterial Series matTag? tag1? tag2? ...
uniaxialMaterial Hardening matTag? E? sigmaY? H_iso? H_kin?
uniaxialMaterial Concrete01 matTag? fpc? epsc0? fpcu? epscu? <-min min?> <-max max?>
5. OpenSees Examples Manual

Open System for Earthquake Engineering Simulation

Examples Manual

Silvia Mazzoni, Frank McKenna, Gregory L. Fenves

Pacific Earthquake Engineering Research Center

University of California, Berkeley

OpenSees version 1.7.3

December 2006

Portable Files:

Example Files can be downloaded from NEEScentral.

The following is a Zipped .chm file and its components:

- Note 1. Windows Operating system only
- Note 2. The links to the examples files do not work, but are included in a subdirectory

OpenSeesExamplesManual.chm.zip
How to Download OpenSees
http://opensees.berkeley.edu

OpenSees 2.3.2 Released

Version 2.3.2 of the OpenSees binary is now available for download. Here is the change log.

Discovering OpenSees

The next seminar in the web-based Discovering OpenSees: Surfing the waves of OpenSees learning series will occur November 17, 2011. This session is titled: OpenSees and Output and will occur on November at 4:30 PM Pacific Time. For more information about registration for this event can be found here.

OpenSees in the Clouds!

OpenSees 2.2.2 is now available for use to all on NEEShub through the OpenSeesLab tool. For those of you with large models or many runs to perform, the machines on which this tool runs is very very very fast.

I have posted a YouTube video showing how to perform an OpenSees simulation using this tool. An improved version will be available soon. The OpenSees GUI applications, BuildingTD and OS Navigator, are also available to run.

Try them out, registration and use of the machines are free!

Welcome

Welcome to the website for OpenSees, a software framework for developing applications to simulate the performance of structural and geotechnical systems subjected to earthquakes.

The goal of the OpenSees development is to improve the modeling and computational simulation in earthquake engineering through open-source development.

OpenSees is under continual development, so users and developers should expect changes and updates on a regular basis. In this sense, all users are developers so it is important to register. More information on Open Source is available.

The development and application of OpenSees is sponsored by the Pacific Earthquake Engineering Research Center through the National Science Foundation engineering and education centers program.

OpenSees has been selected as the simulation component for the George E. Brown, Jr. Network for Earthquake Engineering Simulation and has been sponsored by NEES since 2004. Ongoing work to integrate OpenSees into the NEES framework includes a web-based portal for simulation services, access to the NEESr data repository, and a visualization tool.

More info...
OpenSees Download

Current version is: 2.3.2

To download the code you must be a registered user, and you must enter your email in the box below. Registration is free. We keep track of users and their downloads, so that we can inform them when new updates become available.

Registered User

E-Mail: hadi.kianarang@gmail.com

Submit  Reset

New users must go to the message board registration page.
OpenSees Executable Distribution

Current version is: 2.3.2

No last download is found

OpenSees executables for Windows 98/2000/NT/XP/Vista are available for download. The current version of OpenSees has been tested and is generally stable. However, users may encounter problems when running a new problem for the first time. For that reason we strongly encourage you to participate in the various message boards hosted by OpenSees. And please report any bugs you find. That, of course, is the whole reason why we make these binaries available.

OpenSees uses Tk/Tcl, a general purpose scripting language that we have extended with commands for OpenSees. It is necessary to download a DLL for the Tc/Tk interpreter.

The first step is download the two files below. The first file a zip file containing the OpenSees executable. The second file is a self-installing executable for Tc/Tk.

Note that for those of you who have downloaded before, YOU WILL HAVE TO INSTALL Tk/Tk LIBRARIES AND HEADER FILES AGAIN. This is because we have upgraded to Tk/Tk Version 8.3.8.

<table>
<thead>
<tr>
<th>Windows Binaries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release_2.3.2</td>
</tr>
<tr>
<td>OpenSees2.3.2.exe</td>
</tr>
<tr>
<td>TcTk 8.5.11</td>
</tr>
</tbody>
</table>

After downloading the Tc/Tk executable you will need to run it to install the DLL’s on your computer. As can be seen in the downloading section of the Getting Started Manual you will be asked were to install the files. Currently the default is C:\tcl. It is essential that you change this to “C:\Program Files\Tk” during the course of the installation. If you start OpenSees, you see an error message to the effect, “Cannot find tcl85.dll”, you have skipped this step and must reinstall tcl. Note that you will probably have to uninstall the version you just installed first.

Finally, locate the openase.exe in a convenient directory. It is advisable to execute OpenSees from a DOS shell and you are ready to go.

Mac Version

An OpenSees executable for Apple Machines with Intel processor(s) running OS10.4 or greater is available for download. You can download it here.

OpenSees Tk Applications

OpenSees Tk executables are also available for Windows and Apple Machines with Intel processor(s) running Mac OS10.4 or greater. These applications allow you to build GUIs for OpenSees using the Tk graphical toolkit. The windows application is available here and the Mac application here.
OpenSees: Open Source

Resources for Developers

Welcome! This page contains some useful information for you brave souls who wish to get involved in the code development of OpenSees.

Documentation

Before you begin and for when you get stuck there is always the documentation. For new users to OpenSees, have a look at the primers to get yourselves more familiar with the overall design. For you programmers who need to understand the inner workings of the classes have a look at the Class Specifications.

Download

Download the source via FTP from our server. Details are on the Download page. Source drops to the FTP server usually occur monthly!!.

Builds

Look at the build instructions to find out how to compile this beast on your platform. If you are working on a new platform and get the beast to run,

Contribute

To contribute code, submit your changes to following the instructions. If the code changes are approved they'll be committed.

Browse the Source Code

Browse the up-to-the-minute latest version of the source code online.

OpenSees Help

Click on a directory to enter that directory. Click on a file to display its rev to get a chance to display diffs between revisions.

Current directory: [local] / OpenSees / SRC

<table>
<thead>
<tr>
<th>File</th>
<th>Rev.</th>
<th>Age</th>
<th>API</th>
<th>Last log entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Directory</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attic/ [Don't hide]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>actor/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>analysis/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>convergenceTest/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>coordTransformation/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>damage/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>database/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>doc/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>domain/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>element/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>graph/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>handler/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>machine/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>material/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>matrix/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>modelbuilder/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>nDarray/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>optimization/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>package/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>recorder/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reliability/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>remote/</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### OpenSees Community Forums

- **OpenSees Community Forums**

You last visited on Mon Aug 25, 2008 1:54 pm
The time now is Mon Aug 25, 2008 2:42 pm
The OpenSees Community Forum Index

View posts since last visit
View unmentioned posts

#### Forum
<table>
<thead>
<tr>
<th>Forum</th>
<th>Topics</th>
<th>Posts</th>
<th>Last Post</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenSees</td>
<td>1873</td>
<td>6750</td>
<td>Mon Aug 25, 2008 10:14 am</td>
</tr>
<tr>
<td>OpenSees.exe Users</td>
<td>44</td>
<td>233</td>
<td>Fri Aug 15, 2008 13:42 pm</td>
</tr>
<tr>
<td>Framework</td>
<td>16</td>
<td>14</td>
<td>Wed Jul 23, 2008 10:11 pm</td>
</tr>
<tr>
<td>Parallel Processing</td>
<td>1873</td>
<td>6750</td>
<td>Mon Aug 25, 2008 10:14 am</td>
</tr>
<tr>
<td>Useful Scripts</td>
<td>1873</td>
<td>6750</td>
<td>Mon Aug 25, 2008 10:14 am</td>
</tr>
<tr>
<td>Documentation</td>
<td>1873</td>
<td>6750</td>
<td>Mon Aug 25, 2008 10:14 am</td>
</tr>
<tr>
<td>Future Directions</td>
<td>1873</td>
<td>6750</td>
<td>Mon Aug 25, 2008 10:14 am</td>
</tr>
</tbody>
</table>

Registered login required
Log out [ alivia ] Profile Memberlist FAQ Search

You last visited on Mon Aug 25, 2008 1:54 pm
The time now is Mon Aug 25, 2008 2:42 pm
# Seven Forum Categories

<table>
<thead>
<tr>
<th>Forum</th>
<th>Topics</th>
<th>Posts</th>
<th>Last Post</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OpenSees</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OpenSees.exe Users</strong></td>
<td>1873</td>
<td>6750</td>
<td>Mon Aug 25, 2008 10:14 am</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">TWI</a></td>
</tr>
<tr>
<td><strong>Soil Modelling</strong></td>
<td>16</td>
<td>44</td>
<td>Mon Jul 21, 2008 7:41 pm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">riaeger</a></td>
</tr>
<tr>
<td><strong>Framework</strong></td>
<td>233</td>
<td>829</td>
<td>Fri Aug 15, 2008 11:42 pm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">neallee</a></td>
</tr>
<tr>
<td><strong>Parallel Processing</strong></td>
<td>14</td>
<td>33</td>
<td>Wed Jul 23, 2008 10:11 pm</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">hongqi</a></td>
</tr>
<tr>
<td><strong>Useful Scripts</strong></td>
<td>27</td>
<td>78</td>
<td>Tue Aug 12, 2008 12:15 am</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">khili83</a></td>
</tr>
<tr>
<td><strong>Documentation</strong></td>
<td>127</td>
<td>352</td>
<td>Mon Aug 04, 2008 8:22 am</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">pennycorp56</a></td>
</tr>
<tr>
<td><strong>Future Directions</strong></td>
<td>29</td>
<td>104</td>
<td>Mon Aug 18, 2008 1:18 am</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><a href="#">Andrew</a></td>
</tr>
<tr>
<td>Topics</td>
<td>Replies</td>
<td>Author</td>
<td>Views</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>---------</td>
<td>----------</td>
<td>-------</td>
</tr>
<tr>
<td>Sticky: First Public Release of BuildingTcl</td>
<td>7</td>
<td>silvia</td>
<td>466</td>
</tr>
<tr>
<td>Sticky: OpenSees Days 2008, 8-9 September.</td>
<td>4</td>
<td>silvia</td>
<td>433</td>
</tr>
<tr>
<td>Registration open</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Is consecutive multiple analysis possible in</td>
<td>8</td>
<td>TWI</td>
<td>130</td>
</tr>
<tr>
<td>OpenSees?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>axial load in pushover analysis and IDA method</td>
<td>1</td>
<td>sheng0122</td>
<td>55</td>
</tr>
<tr>
<td>how to move the load pattern?</td>
<td>5</td>
<td>zhmkitten</td>
<td>60</td>
</tr>
<tr>
<td>dynamic analysis of rocking frame</td>
<td>1</td>
<td>ca493</td>
<td>20</td>
</tr>
<tr>
<td>fracture modeling</td>
<td>7</td>
<td>ca493</td>
<td>160</td>
</tr>
<tr>
<td>Sorry...</td>
<td>4</td>
<td>ik295</td>
<td>136</td>
</tr>
<tr>
<td>eigenvalue analysis error</td>
<td>2</td>
<td>mratheor</td>
<td>55</td>
</tr>
<tr>
<td>using scale factor</td>
<td>6</td>
<td>ik295</td>
<td>74</td>
</tr>
<tr>
<td>Model of tallbuilding in opensees</td>
<td>1</td>
<td>dinochen1983</td>
<td>45</td>
</tr>
<tr>
<td>nonlinear static analysis</td>
<td>1</td>
<td>susan</td>
<td>28</td>
</tr>
</tbody>
</table>
Our OpenSees Facebook Group
OpenSees.exe

- OpenSees is an Open-Source Software Framework for developing Nonlinear Finite Element Applications for both sequential and parallel environments.

- OpenSees.exe is an extension of the Tcl interpreter for finite element analysis which uses this framework. It is an example of an application that can be developed using the framework.
What is Tcl

- Tcl is a string-based scripting language.
- Variables and variable substitution
- Expression evaluation
- Basic control structures (if, while, for, foreach)
- Procedures
- File manipulation
- Sourcing other files
WELCOME TO ACTIVETCL

ActiveTcl is ActiveState’s quality-assured distribution of Tcl. The latest Windows, Linux and Mac OS X builds are available for free to the community. Solaris, HP-UX and AIX builds are available in ActiveTcl Business, Enterprise and OEM editions.

ActiveState is committed to making Tcl easy to install and use on all major platforms. This release of ActiveTcl brings you the most stable release of Tcl available in binary form. It also includes several of the most popular extensions pre-compiled and ready to use.

The complete ActiveTcl package contains:

- The binary of the core Tcl distribution
- Popular extensions, pre-compiled
- Self-extracting archives for all platforms
- Complete online documentation

ActiveState’s Tcl Productivity Tools

Check out ActiveTcl Pro Studio, which includes:

- Tcl Dev Kit, the essential toolkit for Tcl programmers. Tcl Dev Kit includes:
  - Tcl Dev Kit Debugger. A convenient graphical user interface that allows you to debug remote and embedded Tcl applications as well as local ones.
  - Tcl Dev Kit Checker. A static code analyzer that helps you find syntax errors and other common usage errors quickly, without having to run your program. Tcl Dev Kit Checker makes it easy to update legacy Tcl code.
  - Tcl Dev Kit Wrapper. Distribute Tcl programs as freestanding executables containing everything needed to run the application.
  - Tcl Dev Kit Compiler. Protect your intellectual property by compiling your Tcl script into a bytecode representation for distribution.
  - ...and more.
- Komodo, ActiveState’s cross-platform, multi-language Integrated Development environment, optimized for open source technologies including Tcl, Perl, PHP, Python, and XSLT.

Tcl Support

- For discussions regarding the ActiveTcl distribution, see the ActiveTcl Mailing List.
- To report ActiveTcl bugs, please see the ActiveTcl Bug Database.
- For general Tcl discussions, see the list of Tcl mailing lists.
How to Install

- Install ActiveTcl8.4.6.1 and ActiveTcl8.5.11 on drive C:\
- Install Install_OSP.exe on drive C:\
- Copy tcleditor folder to C:\
- Go to C:\tcleditor\bin right click on TclEditor.exe select send shortcut to desktop
- Now you can easily use OpenSees by double clicking tcleditor on your desktop
Welcome to SUT OpenSees Pack Installer

This wizard will guide you through the installation of SUT OpenSees Pack.

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

by: Hadi Kenarangi hadi_kenarangi@alum.sharif.edu
Seyed Mojtaba Hosseini Gelekolai hosseinigelekolai@gmail.com

Copyright © 2012, SUT OpenSees Student Group
Hello World! (My First Code)

puts "Hello World!"
GUIs are possible

1. BuildingTcl
BuildingTclViewer: Results - RC Frame

Curvature

Shear Force

Axial Force

Axial Strain
2. OpenSees Navigator

OpenSees Navigator
3. OpenSees Post Processor (OSP)
4. OpenSeesPL (Soil and Pile Modeling)
Finite Element Analysis Softwares

• **Micro modeling**
  ABAQUS, ANSYS, DIANA
  NASTRAN, OPENSEES

• **Macro Modeling**
  ETABS, SAP2000, PERFORM, DRAIN, NONLIN-Pro, IDARC, OPENSEES
How to Compute Performance-Based Deformation Demands?

Increasing Value of Information

- Linear Static Analysis
- Linear Dynamic Modal Response Spectrum Analysis
- Linear Dynamic Modal Response History Analysis
- Linear Dynamic Explicit Response History Analysis

- Nonlinear Static “Pushover” Analysis
- Nonlinear Dynamic Explicit Response History Analysis

✗ = Not Reliable in Predicting Damage

DRAIN-2Dx is old technology, but it represents the basic state of the practice. The state of the art is being advanced through initiatives such as PEER’s OpenSees Environment. (Reference: FEMA technical report 15-5-a)
Why OpenSees

• **Advantages:**
  - Very Fast => Timesaver
  - Open-source => Adding Some Codes
  - Free License => Easy Paper Submission
  - Text File Output => Little H.D.D. Space Consumption
  - Flexible Programming (TCL) => Easy Parametric Studies
  - Very Strong Analysis Engine => Easy Nonlinear Analysis

• **Disadvantages:**
  - No fully developed pre or post processors yet available for model development and visualization
  - Code is under development and still being fine-tuned
Bridge-Soil-Pile Modeling

Elevation and Plan View of Residual Deformation (Scaling Factor = 50)

Numerical Analysis

And Why do Finite Element Analysis NCEER frame tested at the Taiwan facility
Buffalo Shaking Table Test
Objective of User Workshop

• Describe modeling and analysis capability, including: element, section, material.

• Overview of applications, structural not geotechnical.

• Show specific examples of nonlinear analysis.

• Motivation to use OpenSees for your simulation problems.....
What Should be Your Expectations?

• OpenSees is a research tool at this time, but fairly stable for regular use.

• As with any nonlinear analysis, it requires careful consideration of model and interpretation of results.

• It is under continual development by students, faculty and other researchers.

• It is not bullet-proof.

• An investment of time and learning is required.

• The OpenSees open-source community requires contributions for the community to succeed.
Any Questions or Statements?
Basic Modeling
What is Tcl

- Tcl is a dynamic programming language.
- It is a string based command language.
- Variables and variable substitution
- Expression evaluation
- Basic control structures (if, while, for, foreach)
- Procedures
- File manipulation
- Sourcing other files.

Command syntax:

```
command arg1 arg2 ...
```

- Help
Tcl Scripting

• **Variables** and variable substitution

> set a 1
1
> set b a
a
> set b $a
1
> set b $a$a
11

• **Expression** Evaluation

> expr 2+3
5
> set b [expr 2+$a]
3
> set b [expr 2+$a.$a]
3.1

• **Lists**

> set a {9 i c c e}
9 i c c e
> set La [llength $a]
5
> set a_0 [lindex $a 0]
9
> lappend a OpenSees
9 i c c e OpenSees

• **file manipulation**

> set txt [open temp.out w+]
file1792158
> puts $txt “Hello World”
> close $txt
> type temp.out
Hello World

• **procedures & control structures**

> for {set i 1} {$i < 10} {incr i 1} {
puts “i equals $i”
}
> foreach i {9 i c c e} {
puts “i is $i”
}
> proc comp {a b} {
if {$a < $b} {
puts “$a is lower than $b”
} else {
if {$a > $b} {
puts “$a is greater than $b”
} else { puts “$a is equal $b”}
}

> source example.tcl
OpenSees.exe

• There is no GUI!
Example 1 - Eigen Value Problem

$I = 1.943e - 5 \, m^4$

$A = 2.85e - 3 \, m^2$

$E = 2.11 \, \frac{N}{m^2}$

What is First Modal Period?
ModelBuilder Command

Basic ModelBuilder

model Basic -ndm $ndm <ndf $ndf>

<table>
<thead>
<tr>
<th>2D Model:</th>
<th>3D Model:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ndm = 2</td>
<td>ndm = 3</td>
</tr>
<tr>
<td>ndf = 2 or 3</td>
<td>ndf = 3 or 6</td>
</tr>
</tbody>
</table>
Example 1 - Eigen Value Problem

wipe

model basic -ndm 2 -ndf 3
Modeling Commands

Domain

- Single-Point Constraints
  - `fix $nodeTag (ndf $ConstrValues)`

- Geometric Transformation
  - Linear, PDelta, Corotational

- Multi-Point Constraints
  - `equalDOF`, `rigidDiaphragm`, `rigidLink`

Element in Global System

Element in Basic System

```
node $nodeTag (ndm $coords)

mass $nodeTag (ndf $MassValues)

geomTransf Linear $transfTag
```
Example 1 - Eigen Value Problem

wipe

model basic -ndm 2 -ndf 3

node 1 0.0
node 2 3.0
node 3 0.3
node 4 3.3

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000.0 0.0
mass 4 2000.0 0.0

geomTransf Linear 1
Modeling Commands

- Materials

  Uniaxial, nD Material, Section

Uniaxial
- Elastic
- ElasticPP
- Hardening
- Concrete
- Steel
- Hysteretic
- PY-TZ-QZ
- Parallel
- Series
- Gap
- Fatigue
- etc.

nD
- Elastic
- J2
- DruckerPrager
- TemplateElasto-Plasto
- FluidSolidPorous
- PressureMultiYield (dependent, independent)
- etc.

Section
- Elastic
- Fiber

(over 250 material classes)
Modeling Commands

Materials

Uniaxial

uniaxialMaterial Elastic $matTag $E

uniaxialMaterial Concrete01 $matTag $fpc $epsc0 $fpcu $epsU

uniaxialMaterial Steel01 $matTag $Fy $E0 $b
Modeling Commands

- Elements
  - Truss, Elastic Beam Column, Zero Length, Nonlinear Beam Column (force, displacement), Beam With Hinges, Quad, Shell, Brick, Joint, etc.
  - > 100 element classes

```
  element truss $eleTag $iNode $jNode $A $matTag

  element elasticBeamColumn $eleTag $iNode $jNode $A $E $Iz $transfTag

  element nonlinearBeamColumn $eleTag $iNode $jNode $numIntgrPts $secTag $transfTag
```
Example 1-Eigen Value Problem

wipe

model basic -ndm 2 -ndf 3

node 1 0.0.
node 2 3.0.
node 3 0.3.
node 4 3.3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000.0.0.
mass 4 2000.0.0.

geomTransf Linear 1

element elasticBeamColumn 1 1 3 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 2 2 4 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 3 3 4 28.5e-4 2e11 1.943e-5 1
Modeling Commands

• **Eigen Command**

```
eigen $numEigenvalues
```

wipe

g geomTransf Linear 1

```
model basic -ndm 2 -ndf 3

node 1 0.0.
node 2 3.0.
node 3 0.3.
node 4 3.3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000.0 0.
mass 4 2000.0 0.

```
Example 2-1-Gravity Loading

\[ I = 1.943e - 5 \, m^4 \]
\[ A = 2.85e - 3 \, m^2 \]
\[ E = 2. e11 \, \frac{N}{m^2} \]

Total Displacement of Node 4?
Modeling Commands

- Load Pattern
  - Plain, Uniform Excitation, Multi Support

- Plain
- Uniform Excitation
- Multi Support

- Nodal Load
- Elemental Load
  - BeamPointLoad
  - BeamUniformLoad
  - BeamTempLoad

- Acceleration
- Imposed Motion

- SP_Constraint
Modeling Commands

- **Time Series (Functions)**
  - Constant, Linear, Rectangular, Sine, Path

- Series -dt dt? -values {list of points} <-factor cFactor?>
- Series -time {list of times} -values {list of points} <-factor cFactor?>
- Series -dt dt? -filePath fileName? <-factor cFactor?>
- Series -fileTime fileName1? -filePath fileName2? <-factor cFactor?>
Example 2-1-Gravity Loading

wipe

model basic -ndm 2 -ndf 3

node 1 0. 0.
note 2 3. 0.
note 3 0. 3.
note 4 3. 3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000. 0. 0.
mass 4 2000. 0. 0.
geomTransf Linear 1

element elasticBeamColumn 1 1 3 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 2 2 4 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 3 3 4 28.5e-4 2e11 1.943e-5 1

pattern Plain 1 Linear {
  eleLoad -ele 3 -type -beamUniform -5e4
  load 3 3.e5 0. 0.
}
Output Options

There is NO OUTPUT until you request it from OpenSees!

There are 4 options to obtain output:

1. **recorder** command
   Records a specific output to a file or database
   ```
   recorder $type $arg1 $arg2 ...
   
   ```

2. **puts** command
   Puts a specific output or variable to monitor or file stream
   ```
   puts <$fileID> $string
   
   ```

3. **print** command
   Prints a specific output or data existed in the domain to monitor or file stream
   ```
   print <-file $fileName> <-node $nd1 $nd2 ..> <-ele $ele1 $ele2 ..>
   
   ```

4. **recorder display** command
Output Options

Node/EnvelopeNode Recorders

```
recorder Node <-file $fileName> <-timeSeries StsTag> <-time> <-node Stg1 Stg2 ...> -dof $d1 $d2 .. disp
               <-xml $fileName>
               <-binary $fileName>
               <-tcp SinetAddr>
```

• The EnvelopeNode takes exactly same args as Node

```
recorder EnvelopeNode <-file $fileName> <-timeSeries StsTag> <-time> <-node Stg1 Stg2 ...> -dof $d1 $d2 .. disp
               <-xml $fileName>
               <-binary $fileName>
               <-tcp SinetAddr>
```
Output Options

Element/EnvelopeElement Recorders

```
recorder Element <-file $fileName> <-time> <-ele $tg1 $tg2 ...> $arg1 $arg2 ...
```

The valid args for different elements

- **Elastic BCE:**
  - `force`

- **Force BCE and BWHE:**
  - `force`
  - `globalForce`
  - `localForce`
  - `plasticDeformation`
  - `etc.`

- The EnvelopeElement takes exactly same args
Example 2-1-Gravity Loading

wipe

model basic -ndm 2 -ndf 3

node 1 0.0.
nodel 2 3.0.
nodel 3 0.3.
nodel 4 3.3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000.0.0.
mass 4 2000.0.0.
geomTransf Linear 1

element elasticBeamColumn 1 1 3 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 2 2 4 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 3 3 4 28.5e-4 2e11 1.943e-5 1

pattern Plain 1 Linear {
  eleLoad -ele 3 -type -beamUniform -5e4
  load 3 3.e5 0.0.
}

recorder Node -file node1disp.out -time -node 4 -dof 1 2 3 disp
recorder Node -file node1reac.out -time -node 1 -dof 1 2 3 reaction
recorder Node -file node2reac.out -time -node 2 -dof 1 2 3 reaction
Example Analysis

**Static Linear Analysis with Load Control**

- constraints Plain
- numberer Plain
- system BandGeneral
- test NormDispIncr 1.e-8 6
- algorithm ModifiedNewton
- integrator LoadControl 1
- analysis Static
- analyze 1
- loadConst -time 0.0

**Static Nonlinear Analysis with Load Control**

- constraints Plain
- numberer Plain
- system BandGeneral
- test NormDispIncr 1.e-8 6
- algorithm ModifiedNewton
- integrator LoadControl 0.1
- analysis Static
- analyze 10
- loadConst -time 0.0
Example 2-1-Gravity Loading

wipe

model basic -ndm 2 -ndf 3

node 1 0. 0.
nod 2 3. 0.
nod 3 0. 3.
nod 4 3. 3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000. 0. 0.
mass 4 2000. 0. 0.
gemTransf Linear 1

element elasticBeamColumn 1 1 3 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 2 2 4 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 3 3 4 28.5e-4 2e11 1.943e-5 1

pattern Plain 1 Linear {
    eleLoad -ele 3 -type -beamUniform -5e4
    load 3 3.e5 0. 0.
}

recorder Node -file node4disp.out -time -node 4 -dof 1 2 3 disp
recorder Node -file node1reac.out -time -node 1 -dof 1 2 3 reaction
recorder Node -file node2reac.out -time -node 2 -dof 1 2 3 reaction

constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr 1.e-8 6
algorithm ModifiedNewton
integrator LoadControl 1
analysis Static
analyze 1
loadConst -time 0.0
Example 2-2-Linear Pushover

Push the frame to 0.1m displacement of Node 4 in X dir.
Example Analysis

**Static Linear Analysis with Displacement Control**

constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr 1.e-8 6
algorithm ModifiedNewton
integrator DisplacementControl 4 1 0.001
analysis Static
analyze 100
loadConst -time 0.0
Example 2-2-Linear Pushover

```plaintext
wipe

model basic -ndm 2 -ndf 3
node 1 0. 0.
node 2 3. 0.
node 3 0. 3.
node 4 3. 3.
fix 1 1 1 1
fix 2 1 1 1
mass 3 2000. 0. 0.
mass 4 2000. 0. 0.
geomTransf Linear 1

element elasticBeamColumn 1 1 3 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 2 2 4 28.5e-4 2e11 1.943e-5 1
element elasticBeamColumn 3 3 4 28.5e-4 2e11 1.943e-5 1

pattern Plain 1 Linear {
  load 3 1000. 0. 0.
}
```
Any Questions or Statements?
Nonlinear Modeling and Analysis
Why Nonlinear Analysis:

- Geometric Nonlinearities
- Material nonlinearities
- Contact nonlinearities
Nonlinear Analysis is Harder

- It requires much more thought when setting up the model
- It requires more thought when setting up the analysis
- It takes more computational time.
- It does not always converge.
- It does not always converge to the correct solution.

BUT Most Problems Require
Nonlinear Analysis
CHECK YOUR MODEL!
Modeling Commands

• **Section Command**

What is a section?
A section defines the stress resultant force-deformation response at a cross section of a beam-column or plate element.

**Types of sections:**

- Elastic
- Resultant
- Fiber
Modeling Commands

- **Fiber Section**
  The Fiber Section object is composed of Fiber objects.

```
section Fiber $secTag {
  fiber <fiber arguments>
  patch <patch arguments>
  layer <layer arguments>
}
```
Modeling Commands

- Fiber command

```
fiber $yLoc $zLoc $A $matTag
```
Modeling Commands

- Patch command

```plaintext
patch quad $matTag $numSubdivIJ $numSubdivJK $yI $zI $yJ $zJ $yK $zK $yL $zL

patch circ $matTag $numSubdivCirc $numSubdivRad $yCenter $zCenter $intRad $extRad <$startAng $endAng>
```
Modeling Commands

- Layer command

```plaintext
layer straight $\text{matTag} \enspace \text{numBars} \enspace \text{areaBar} \enspace \text{yStart} \enspace \text{zStart} \enspace \text{yEnd} \enspace \text{zEnd}
```

```plaintext
layer circ $\text{matTag} \enspace \text{numBar} \enspace \text{areaBar} \enspace \text{yCenter} \enspace \text{zCenter} \enspace \text{radius} \enspace <$\text{startAng} \enspace \text{endAng}$>
```
Fiber Section Example

section Fiber 1 {
    yI  zI  yJ  zJ  yK  zK  yL  zL
    patch quad 1 2 8 -0.15  0.125 -0.15 -0.125  0.15 -0.125  0.15 0.125
}

Diagram showing a fiber section with coordinates and dimensions.
Push the frame to 0.1m displacement of Node 4 in X dir.
Example 3-NonLinear Pushover

```plaintext
wipe

model basic -ndm 2 -ndf 3

node 1 0 0.
node 2 3 0.
node 3 0 3.
node 4 3 3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000. 0. 0.
mass 4 2000. 0. 0.

geomTransf Linear 1

uniaxialMaterial Steel01 1 2.354e8 2.e11 0.02
```
Example 3-NonLinear Pushover

wipe

model basic -ndm 2 -ndf 3

node 1 0.0.
node 2 3.0.
node 3 0.3.
node 4 3.3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000.0.0.
mass 4 2000.0.0.

geomTransf Linear 1

uniaxialMaterial Steel01 1 2.354e8 2.e11 0.02

section fiber 1 {
patch quad 1 2 3 0.0915 0.05 0.0915 -0.05 0.1 -0.05 0.1 0.05
patch quad 1 2 8 -0.0915 0.0028 -0.0915 -0.0028 0.0915 -0.0028 0.0915 0.0028
patch quad 1 2 3 -0.1 0.05 -0.1 -0.05 -0.0915 -0.05 -0.0915 0.05
}

\[ bf = 10cm \]
\[ d = 20cm \]
\[ tf = 0.85cm \]
\[ tw = 0.56cm \]
Example 3-NonLinear Pushover

wipe

model basic -ndm 2 -ndf 3

node 1 0. 0.
node 2 3. 0.
node 3 0. 3.
node 4 3. 3.

fix 1 1 1 1
fix 2 1 1 1

mass 3 2000. 0. 0.
mass 4 2000. 0. 0.

geomTransf Linear 1

uniaxialMaterial Steel01 1 2.354e8 2.e11 0.02

section fiber 1 {
  patch quad 1 2 3 0.0915 0.05 0.0915 -0.05 0.1 -0.05 0.1 0.05
  patch quad 1 2 8 -0.0915 0.0028 -0.0915 -0.0028 0.0915 -0.0028 0.0915 0.0028
  patch quad 1 2 3 -0.1 0.05 -0.1 -0.05 -0.0915 -0.05 -0.0915 0.05
}

element nonlinearBeamColumn 1 1 3 10 1 1
element nonlinearBeamColumn 2 2 4 10 1 1
element nonlinearBeamColumn 3 3 4 10 1 1

pattern Plain 1 Linear {
  load 3 10. 0. 0.
}

recorder Node -file node4disp.out -time -node 4 -dof 1 2 3 disp
recorder Node -file node1reac.out -time -node 1 -dof 1 2 3 reaction
recorder Node -file node2reac.out -time -node 2 -dof 1 2 3 reaction
Example 3 (Analysis)

Static Nonlinear Analysis with Displacement Control

constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr 1.e-8 6
algorithm ModifiedNewton
integrator DisplacementControl 4 1 0.001
analysis Static
analyze 100
loadConst -time 0.0
Example 4-1-NonLinear Earthquake-Without Damping

\[ b_f = 10\, cm \]
\[ d = 20\, cm \]
\[ t_f = 0.85\, cm \]
\[ t_w = 0.56\, cm \]

Nonlinear Time History Analysis
Example 4-1-NonLinear Earthquake-Without Damping

- Uniform Excitation Pattern command

```
pattern UniformExcitation $patternTag $dir -accel (TimeSeriesType arguments) <-vel0 $ver0>
```

```plaintext
set accel "Series -dt 0.02 -filePath TabasLN(dt=0.02)PGA=0.4g.txt -factor [expr 9.81]"

pattern UniformExcitation 3 1 -accel $accel
```

**Time History Nonlinear Analysis**

```plaintext
set dt 0.02

constraints Plain
numberer Plain
system BandGeneral
test NormDispIncr 1.0e-8 10
algorithm Newton
integrator Newmark 0.5 0.25
analysis Transient
analyze [expr int(32.82/$dt)] $dt
```
Example 4-2-NonLinear Earthquake-With Damping

\[ bf = 10cm \]
\[ d = 20cm \]
\[ tf = 0.85cm \]
\[ tw = 0.56cm \]
\[ \xi = 0.05 \]

Nonlinear Time History Analysis
Example 4-2-NonLinear Earthquake-With Damping

- **rayleigh command**

```plaintext
rayleigh $\alphaM$ $\betaK$ $\betaKinit$ $\betaKcomm$
```

\[
\alpha = \frac{2\xi\omega_i\omega_j}{\omega_i + \omega_j}
\]

\[
\beta = \frac{2\xi}{\omega_i + \omega_j}
\]

```
rayleigh 1.19 0 0 0.0021
```
Example 4-4-NonLinear Earthquake-With Tuned Mass Damper

\[ M_{tmd} = 80 \, Kg \]
\[ K_{tmd} = 45609.9 \, \frac{N}{m} \]
\[ C_{tmd} = 0 \]
\[ C_{structure} = 0 \]
Example 4-4-NonLinear Earthquake-With Tuned Mass Damper

How to model a simple Tuned Mass Damper:

\[ M_{tmd} = 80 \text{ Kg} \]
\[ K_{tmd} = 45609.9 \frac{N}{m} \]
\[ C_{tmd} = 0 \]

node 5 3.3.
fix 5 0 1 1
mass 5 80.0 0.0.
uniaxialMaterial Elastic 2 45609.9
element zeroLength 4 4 5 -mat 2 -dir 1
Example 4-4-NonLinear Earthquake-With Tuned Mass Damper

Ground Motion Record

Base Shear

Disp. Node 4

Dotted Line: Uncontrolled

Bold Line: Controlled with TMD
Any Questions or Statements?
Earthquake Doesn’t Kill People, Buildings Do!

Thank You.

Nov. 2012